ASSIGNMENT 10 OUTPUT

Create a table named teachers with fields id, name, subject, experience and salary and insert 8 rows.

id	name	subject	experience	salary
1	Alice	Math	5	50000.00
2	Bob	Science	8	60000.00
3	Charlie	History	12	70000.00
4	Dana	English	2	40000.00
5	Evan	Computer Science	15	90000.00
6	Fay	Art	3	45000.00
7	Grace	Physics	9	75000.00
8	Hank	Biology	10	80000.00
NULL	NULL	NULL	NULL	NULL

2. Create a before insert trigger named before_insert_teacher that will raise an error “salary cannot be negative” if the salary inserted to the table is less than zero.

```
32    --  2. Create a before insert trigger named before_insert_teacher that will raise an error “salary cannot
33    DELIMITER $$
34 •  CREATE TRIGGER before_insert_teacher
35    BEFORE INSERT ON teachers
36    FOR EACH ROW
37    BEGIN
38        if new.salary <0 then
39        signal sqlstate '45000' set message_text = 'salary cannot be negative' ;-- default signal sqlstate 45
40        end if;
41        -- Block of code (trigger logic here)
42    END $$
43    DELIMITER ;
44
45 •  SHOW triggers;
46 •  INSERT INTO teachers (name, subject, experience, salary)
47    VALUES ('Ivy', 'Music', 4, -5000);
48
49
```

Output

Action Output

#	Time	Action	Message
342	08:30:28	SHOW triggers	7 row(s) returned
343	08:32:59	INSERT INTO teachers (name, subject, experience, salary) VALUES ('Ivy', 'Music', 4, -5000)	Error Code: 1644. salary cannot be negative

3. Create an after insert trigger named after_insert_teacher that inserts a row with teacher_id,action,

timestamp to a table called teacher_log when a new entry gets inserted to the teacher table. tecaher_id -> column of teacher table,

action -> the trigger action, timestamp -> time at which the new row has got inserted.

```
CREATE TABLE teacher_log (
    log_id INT AUTO_INCREMENT PRIMARY KEY,
    teacher_id INT,
    action VARCHAR(50),
    log_time TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
DELIMITER $$
CREATE TRIGGER after_insert_teacher
AFTER INSERT ON teachers
FOR EACH ROW
BEGIN
    insert into teacher_log(teacher_id,action) values
    (new.id,'Teacher recently joined');
END $$
DELIMITER ;

INSERT INTO teachers (name, subject, experience, salary)
VALUES ('Ivy', 'Music', 4, 50000);

select * from teachers;
select * from teacher_log;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

log_id	teacher_id	action	log_time
1	9	Teacher recently joined	2024-12-08 08:42:17
NULL	NULL	NULL	NULL

4. Create a before delete trigger that will raise an error when you

try to delete a row that has experience greater than 10 years.

```
78  DELIMITER $$
79  CREATE TRIGGER before_delete_teacher
80  BEFORE DELETE ON teachers
81  FOR EACH ROW
82  BEGIN
83      IF old.experience > 10 THEN
84          SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Cannot delete a teacher with more than 10 years of experience';
85      END IF;
86  END $$
87  DELIMITER ;
88
89  DELETE FROM teachers WHERE experience>10;
```

Output

Action Output

#	Time	Action	Message
351	08:48:29	select * from teachers LIMIT 0, 1000	9 row(s) returned
352	08:48:38	DELETE FROM teachers WHERE experience>10	Error Code: 1644. Cannot delete a teacher with more than 10 years of experience

5. Create an after delete trigger that will insert a row to teacher_log table when that row is deleted from teacher table.

```
92  -- 5. Create an after delete trigger that will insert a row to teacher_log table when that row is deleted from teacher table.
93  DELIMITER $$
94  CREATE TRIGGER after_delete_teacher
95  AFTER DELETE ON teachers
96  FOR EACH ROW
97  BEGIN
98      INSERT INTO teacher_log (teacher_id, action)
99      VALUES (OLD.id, 'DELETED');
100 END $$
101 DELIMITER ;
102
103 DELETE FROM teachers WHERE id=1;
104 SELECT * FROM teacher_log;
105
106
```

log_id	teacher_id	action	log_time
1	9	Teacher recently joined	2024-12-08 08:42:17
2	1	DELETED	2024-12-08 08:52:15
NULL	NULL	NULL	NULL